

# Objective C Programming

## Lecture 2

More on OOP. Accessors. Xcode.

International University of Information Technology  
Almaty, 2012

# Creating objects

```
NSString *title = [[NSString alloc] init];
```

vs.

```
NSString *title = [NSString string];
```

```
NSNumber* value = [[NSNumber alloc] initWithFloat:1.0];
```

# Releasing objects

If automatic reference counting is disabled, you need to release memory manually

```
// string1 will be released automatically  
NSString* name = [NSString string];
```

```
// must release this when done  
NSString* song = [[NSString alloc] init];  
[string2 release];
```

# Generating accessors

```
@interface Fraction : NSObject
@property int numerator, denominator;
-(void) print;
-(double) convertToNum;
@end

@implementation Fraction
@synthesize numerator, denominator;
    -(void) print {
        NSLog ("%i/%i", numerator, denominator);
    }
@end

[myFraction setNumerator: 3]
[myFraction numerator]
```

# Multiple arguments

## The nice way

```
[myFraction setNumerator:10];  
[myFraction setDenominator:22];
```

# Multiple arguments

## The nice way

```
[myFraction setNumerator:10];  
[myFraction setDenominator:22];
```

```
[myFraction setNumerator:10 setDenominator:22];
```

# Multiple arguments

## The nice way

```
[myFraction setNumerator:10];  
[myFraction setDenominator:22];
```

```
[myFraction setNumerator:10 setDenominator:22];  
[myFraction setNumerator:10 andDenominator:22];
```

# Multiple arguments

## The nice way

```
[myFraction setNumerator:10];  
[myFraction setDenominator:22];
```

```
[myFraction setNumerator:10 setDenominator:22];  
[myFraction setNumerator:10 andDenominator:22];
```

```
[myFraction setTo:10 over:22];
```



# Let's add something to our class...

# Self

like 'this' in C++  
pointer to the current instance

# Inheritance

- Basically, nothing new for you

# NSObject methods

```
[myFraction isKindOfClass: [Fraction class]];
```

```
[myFraction isKindOfClass: [Fraction class]];
```

```
[myFraction respondsToSelector: @selector (setNumerator:)];
```

```
[Fraction instancesRespondToSelector:@selector (setNumerator:)];
```

```
[Fraction isKindOfClass: [NSObject class]];
```

# Exception handling

```
@try {  
    // something  
}  
  
@catch (NSException *exception) {  
    NSLog(@"Caught %@%@", [exception name], [exception reason]);  
}
```

# Let's make a browser!